SOFTWARE REQUIREMENTS SPECIFICATION (SRS) FOR THE DII COE COMMON MESSAGE PROCESSOR,

JULY 13, 1997

Prepared for:
Program Manager
Common Hardware Software
Ft. Monmouth, NJ 07703

SOFTWARE REQUIREMENTS SPECIFICATION
for the
DEFENSE INFORMATION INFRASTRUCTURE (DII)
COMMON OPERATING ENVIRONMENT (COE)
MESSAGE PROCESSOR

CONTRACT NO: DAAB07-91-D-Q502

Prepared For:

PROJECT MANAGER COMMON HARDWARE AND SOFTWARE
FORT MONMOUTH, NEW JERSEY 07703

Prepared By:
THE ANALYSIS CORPORATION
in cooperation with
TELOS

TABLE OF CONTENTS

FIGURES

TABLES

1. SCOPE

1.1 IDENTIFICATION

This Software Requirements Specification (SRS) defines the software requirements for the Defense Information Infrastructure (DII) Common Operating Environment (COE) message processing functional area. The purpose of the message processing functional area is to provide message receipt, routing, storage, retrieval, parsing, generation, coordination, release and processing of standing request for information for information transfer using messages conforming to the Message Text Format (MTF) rules.

1.2 SYSTEM OVERVIEW

The message processing module is logically bounded on one side by the communications module and on the other side by supported processes and/or other COE modules. Processing of inbound message from the communications front end includes such essential functions as field/format validation, profiling, standing request for information, parsing, and routing. Processing of messages for hand off to the communications front end includes such essential functions as message preparation, validation, header preparation and coordination/release. The message processor is capable of processing both formatted and unformatted messages which are validated by tables derived from the Joint Interoperability Engineering Office (JIEO) Central Data Base System (CDBS). Message processor module components may be employed independently to perform a single, or group of, function(s) such as construct message reports while another tool validates. Figure 1-1 provides a top level functional flow of the message processing module and identifies three major subordinate areas within the processing module. Two of these areas are inbound and outbound processing which contain functionality specific to that process. The third is a support services which contains functionality used to by both inbound and outbound processing.

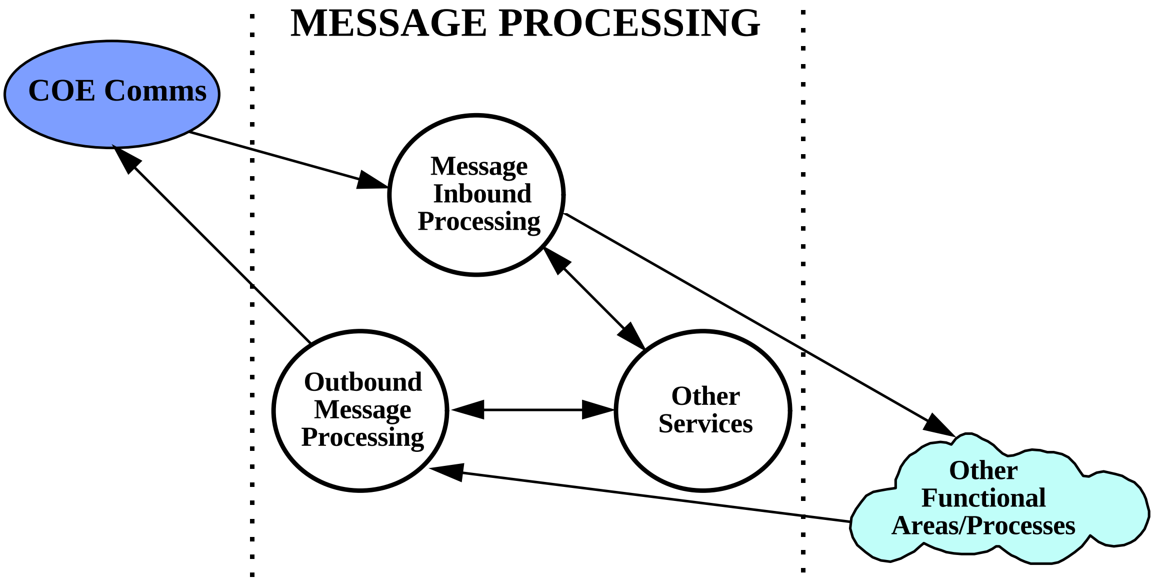
MESSAGE PROCESSING
COE Comms
Message Inbound Processing
Outbound Message Processing
Other Services
Other Functional Areas/Processes

Figure 1-1. Message Processing Module

1.2.1 COMMON OPERATING ENVIRONMENT

The COE is intended for use by all Department of Defense Command and Control Systems (CCS) and Automated Information Systems (AIS) as the infrastructure on which they reside. The COE is defined as an integrated architecture made up of hardware and software components that provides standard, modular, system and applications support software for a tailorable set of functional applications. COE software will be developed for the following platforms, at a minimum:

HP 9000 series using HP/UX 9.07 (RISC)
HP 9000 series using HP/UX 10.10 (RISC)
SUN using SOLARIS 2.4
SUN using SOLARIS 2.51
Windows 3.1x operating systems
Windows NT operating systems

1.2.2 MESSAGE PROCESSING MODULE OVERVIEW

The message processing module provides for message receipt, from a communications front end; internal message routing; the generation, coordination and release of outbound messages; data normalization; storage and retrieval; message profiling; and format validation.

The message processing functional area consist of modularized and callable software that supports message parsing, message storage and retrieval, scanning of inbound messages for satisfaction of Standing Request for Information (SRI), internal routing of messages, message creation (automatically or interactively), data normalization, retrospective search, and error handling. It is a generic, table driven processor that accepts formatted and unformatted MTF like messages from a communications front end, validates message format and field content, then performs additional processing as directed by the user. Figure 1-2 provides a functional flow of the message processing module.

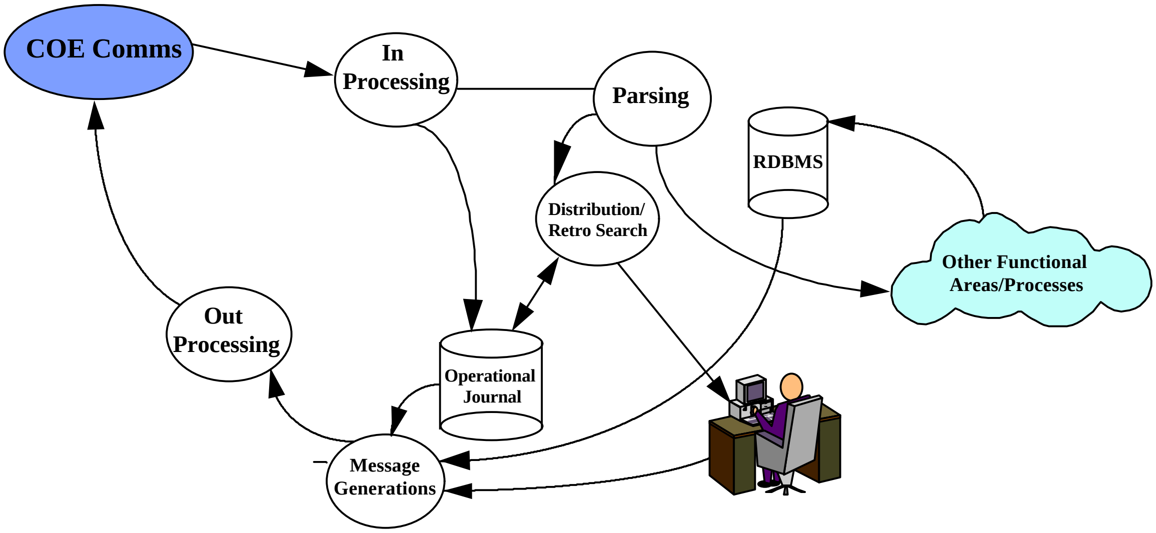

Figure 1-2. Message Processing Module Functional Flow Diagram

1.3 DOCUMENT OVERVIEW

Section 1 of this document identifies the message processing functional area and provides an overview of the capabilities.

A list of applicable government and non-government documents is found in Section 2.

Section 3 details the functional capabilities for the message processing area including the internal and external interfaces, performance requirements, security requirements, and design constraints.

Section 4 is the qualification requirements including the methods and special qualification requirements.

The requirements traceability matrix is located in Section 5 of this document.

2. REFERENCED DOCUMENTS

2.1 GOVERNMENT DOCUMENTS

The following documents form a part of this specification to the extent specified herein. In the event of a conflict between the documents referenced herein and the contents of this specification, the contents of this specification shall be considered a superseding requirement. Latest versions of documents are applicable where exact dates are not provided.

2.1.1 SPECIFICATIONS

(1) MROC 9-81: JCS Multi-Command Required Operational Capability (MROC) for Automated Message Handling, May 1981, Updated July 1983.

2.1.2 STANDARDS

(1) MIL STD 6040 U.S. Message Text Formatting Program, Description of U.S. Message Text Formatting Program

(2) ACCS-A3-500-004 Army Command and Control System Message Catalog, 28 May 1993.

(3) ACCS-A3-500-005 Message Format Definition Database Specification.

(4) Joint Variable Message Format (VMF) Technical Interface Design Plan.

(5) Intelligence and Electronic Warfare (IEW) Character-Oriented Message Catalog (COMCAT)

(6) United States Signals Intelligence Directive (USSID) 316

(7) Marine Tactical System (MTS), Marine Corps Unique Message Standard

(8) OTH-GOLD, Navy Unique Message Standard

(9) ADatP-3, Allied Data Publication

2.1.3 OTHER GOVERNMENT DOCUMENTS

(1) DSSCS Operating Instructions (DOI) 103

(2) Allied Communications Publication (ACP) 126: GENSER Operating Procedures.

(3) Allied Communications Publication (ACP) 126(M): GENSER Operating Procedures (Modified).

(4) Allied Communications Publication (ACP) 127: NATO Operating Procedures.

(5) DD173: Joint Message Form, January 1979.

(6) MIL STD 1832: Diskette Message File Formats for Defense Messaging, 6 Sep 93 with latest change pages.

(7) Allied Communications Publication (ACP) 123.

(8) Allied Communications Publication (ACP) 127(M): NATO Operating Procedures (Modified).

(9) MIL STD 2045-47001, Application Layer Protocol

(10) User Interface Specifications for the Global Command and Control System (GCCS)

(11) Defense Intelligence Agency Manual (DIAM) 65-19

(12) Army Regulation (AR) 380-19, Information Systems Security

(13) Department of Defense (DoD) Standard 5200.28, Department of Defense Trusted Computer System Evaluation Criteria

(14) Joint Chiefs of Staff (JCS) Publication 6-03.7

(15) Joint Army, Navy, and Air Force Protocol (JANAP) 128

3. MESSAGE PROCESSING REQUIREMENTS

3.1 REQUIRED STATES AND MODES

The message processing module shall:

3.1.1 Support all defined DII COE states

3.1.2 Support all defined DII COE modes

3.2 MESSAGE PROCESSING REQUIREMENTS

3.2.1 MESSAGE INBOUND PROCESSING

Inbound processing consist of receiving message packets from the communications front end, routing of the message for proper processing, validation, parsing, checking for SRI satisfaction, information labeling, and handing off data to an external process/COE module for further processing. A flow chart depicting the association of these processes can be seen in figure 3-1.

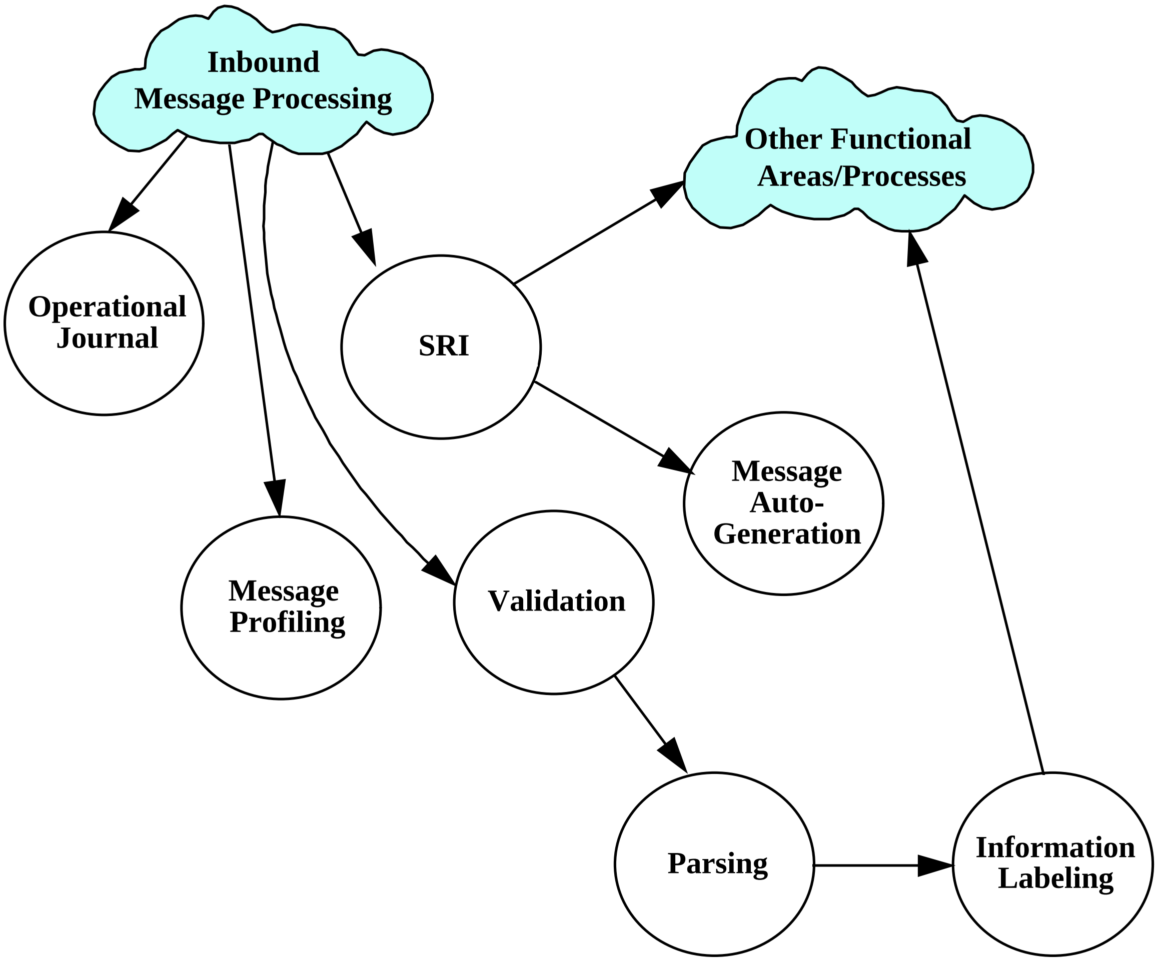

FIGURE 3-1 MESSAGE INBOUND PROCESSING

3.2.1.1 INTERNAL ROUTING

The message processor shall be capable of accepting data from the communications processing functional area and routing the data to various areas, based on processing requirements. Routing shall be to:

(a) Operational Journal
(b) Parser
(c) Profiler
(d) SRI processing
(e) User Interface for error processing

To support routing of data to the appropriate module(s) the message processor shall:

3.2.1.1.1 Provide the capability to extract office symbols from received message headers and route the message to the appropriate destination queue.

3.2.1.1.2 Provide the capability to route based on action addressees

3.2.1.1.3 Provide the capability to route based on information addressees

3.2.1.1.4 Provide the capability to route based on Address Information Group (AIG) routers

3.2.1.1.5 Route to automatic message generation based upon user defined criteria (SRI) which may be satisfied through:

(a) message content
(b) message type
(c) message originator
(d) message classification
(e) message precedence

3.2.1.1.6 Provide the capability to forward received messages to addresses on a secondary distribution list which shall be user definable and maintainable.

3.2.1.1.7 Support routing of messages from a file or application to:

(a) a secondary storage media
(b) an output device (i.e.; tape, printer, etc.)

3.2.1.1.8 Provide the capability to deliver received messages to an address contained in the router table (i.e.: an individual, a group (role), a database, or an application).

3.2.1.1.9 Provide the capability to transfer received messages to their destination (individual, role, database, or application) in order of precedence

3.2.1.1.10 Provide the capability to transfer received messages to their destination (individual, role, database, or application) in order of arrival, FIFO

3.2.1.1.11 Provide the capability to transfer received messages to their destination (individual, role, database, or application) in reverse order of arrival Last In First Out (LIFO).

3.2.1.1.12 Provide the capability to route to designated user for interactive processing upon:

(a) Message having identified errors
(b) Message being identified as an incomplete sectioned message
(c) Message being identified as a textual message
(d) Message identified for manual processing

3.2.1.2 MESSAGE PARSER

The message processing module shall provide users and authorized processes the capability to extract data from a message. The message processing module shall:

3.2.1.2.1 Provide the capability to specify the information to be extracted from a message.

3.2.1.2.2 Provide the ability to locally change the configuration for data extraction requirements of users and authorized processes without affecting data extraction requirements of other users or processes.

3.2.1.2.3 Restrict ability to modify extraction requirements to authorized users, only.

3.2.1.2.4 Identify the presence of validation errors in messages when delivering extracted information to users and to authorized processes.

3.2.1.2.5 Not impose arbitrary restrictions on the quantity of data extracted from a message.

3.2.1.2.6 Deliver received messages in whole or part to files, processes, and databases.

3.2.1.2.7 Parse, in addition to those referenced in paragraph 2.1.2 all message standards that can be translated into a format compatible with the MTF definition rules

3.2.1.2.8 Identify a message as requiring manual processing if any of the following conditions are true:

(a) The message contains detected, uncorrectable errors
(b) The message is a free text message
(c) The message type has been designated for manual processing

3.2.1.2.9 Protect against parsing the same message twice even though the same message (must contain same originator and date-time group) may be received from the communications module on multiple occasions.

3.2.1.2.10 Be capable of identifying a message as requiring interactive correction.

3.2.1.2.11 Make clearly visible invalid entries when presented to a user for action.

3.2.1.2.12 Provide interactive and/or on-line help data specifying valid entries and/or format.

3.2.1.2.13 Support submission of corrected messages for parsing.

3.2.1.2.14 Check and validate for operational or exercise message.

3.2.1.2.15 Provide the capability to selectively expedite the processing of a message in accordance with it’s assigned precedence. Rank order of precedence is:

(a) Emergency Command Precedence (ECP)
(b) Flash
(c) Immediate
(d) Priority
(e) Routine

3.2.1.2.16 Provide the capability for queuing of messages for parsing by precedence.

3.2.1.2.17 Provide the capability for retrieval of messages by precedence.

3.2.1.2.18 Protect data integrity through use of persistent queuing.

3.2.1.2.19 Protect data integrity through use of cyclic redundancy checking (CRC) or like methodology.

3.2.1.2.20 Provide interface to define data extraction criteria.

3.2.1.2.21 Provide interface to edit data extraction criteria.

3.2.1.2.22 Provide interface to activate a data extraction criteria.

3.2.1.2.23 Provide interface to de-activate a data extraction criteria.

3.2.1.2.24 Route parsing results to addresses which are user definable.

3.2.1.2.25 Access data recipient address(s) in a table definable by the system administrator.

3.2.1.2.26 Access data recipient address(s) in a table maintainable by the system administrator.

3.2.1.3 INFORMATION LABELING

For systems operating at the system high mode of operations, the message processor shall support the auto input of an information label at the system high water mark, as required by the following extract from DoD 5200.28-STD which states "System high security mode - The mode of operation in which system hardware/software is only trusted to provide need-to-know protection between users. In this mode, the entire system, to include all components electrically and/or physically connected, must operate with security measures commensurate with the highest classification and sensitivity of the information being processed and/or stored. All system users in this environment must possess clearances and authorizations for all information contained in the system. All system output must be clearly marked with the highest classification and all system caveats, until the information has been reviewed manually by an authorized individual to ensure appropriate classifications and caveats have been affixed." Records created by the parser shall be annotated with an information label commensurate with the level of classification associated with the parsed message (legal classifications are contained in DIAM 65-19). The message processor shall:

3.2.1.3.1 For system high systems, attach an information label to each exported data record equal to the system high water mark. Messages received which are classified less than the system high water mark will be checked for legality and considered legal if the classification marking is equal to or subordinate to the system high water mark.

3.2.1.3.2 For multiple levels of security (MLS) systems the information label must be equal to or subordinate to the system high water mark of the system. Messages received which are classified less than the system high water mark will be checked for legality and considered legal if the classification marking is equal to or subordinate to the system high water mark. Records created from messages under this requirement shall be labeled according to the classification marking of the incoming data.

3.2.1.3.3 User modification of security classification (modification of the classification marking) must be monitored and reported to system audit trail.

3.2.1.3.4 Legal information labels shall be created based upon hierarchical classifications

3.2.1.3.5 Legal information labels shall be created based upon hierarchical plus non-hierarchical classifications

3.2.1.4 SRI PROCESSING

Standing Request for Information (SRI) processing is the process of monitoring incoming messages to detect if they contain information of interest to a user or process. Definition of desired information is termed a "criteria" and is user definable and maintainable. The SRI process shall:

3.2.1.4.1 Provide the capability to initiate a local message-based SRI.

3.2.1.4.2 Provide the capability to initiate a remote message-based SRI.

3.2.1.4.3 Provide the capability to define an activation condition for a message-based SRI.

3.2.1.4.4 Provide the capability to terminate all message-based SRIs managed by an application program with a single action.

3.2.1.4.5 Provide the capability for an application to terminate local message-based SRIs unconditionally.

3.2.1.4.6 Provide the capability for an application to terminate remote message-based SRIs unconditionally.

3.2.1.4.7 Provide the capability to terminate a message-based SRI after one activation.

3.2.1.4.8 Provide the capability to identify all active message-based SRIs (local and remote) to an application.

3.2.1.4.9 Provide the capability to transfer active message-based SRIs to on-line, non-volatile storage during normal W/S termination.

3.2.1.4.10 Provide the capability to restore active message-based SRIs during W/S initialization.

3.2.1.4.11 Provide the capability to monitor incoming messages for the satisfaction of message-based SRI conditions.

3.2.1.4.12 Provide the capability to initiate the required message-based SRI processing

3.2.1.4.13 Provide the capability to notify a local application upon the satisfaction of a message-based SRI.

3.2.1.4.14 Provide the capability to notify a remote application upon the satisfaction of a message-based SRI.

3.2.1.4.15 Provide the capability to route data satisfying a SRI to a specified user or process. Specific users are:

(a) Alerts
(b) Auto message generation
(c) Specified DBMS
(d) Specified user
(e) Process

3.2.1.5 MESSAGE PROFILING

Messages entering the message processing module must be processed to identify and/or extract (copy or demarcate) message elements. Elements extracted from messages are necessary for construction of corresponding message summaries which shall provide "two-level" message review capabilities required for distribution, coordination, and retrieval functions used during search and retrieval processes. The message processing module shall:

3.2.1.5.1 Provide the capability to enter message selection profiles for individual users, group of users, or processes.

3.2.1.5.2 Support distribution criteria to include message routing criteria or message content.

3.2.1.5.3 Compare messages and system profiles to determine which accounts should receive a copy of the associated message summaries.

3.2.1.5.4 Compare messages and system profiles to determine which accounts should receive a copy of the associated parsed data.

3.2.1.5.5 Determine from user profiles and message content which organization is assigned Action for the Office of Primary Interest (ACT/PI) for a given message.

3.2.1.5.6 Determine from user profiles and message content which account shall be designated the Action Officer (AO) for the message.

3.2.1.5.7 Provide an optional capability to distribute message summaries of transmitted messages (come-back copies) to any combination of accounts listed in the internal distribution fields (drafter, coordinators, releaser).

3.2.1.5.8 Provide an optional capability to distribute message come-back copies to any combination of accounts listed in the internal distribution fields (drafter, coordinators, releaser).

3.2.1.5.9 Support user re-addressal of received messages for retransmission to external organizations.

3.2.1.5.10 To support generation of a message profile the message processor shall:

(a) Extract the following precedence markings (on messages so marked):

(1) EMERGENCY COMMAND PRECEDENCE (ECP)
(2) FLASH
(3) IMMEDIATE
(4) PRIORITY
(5) ROUTINE

(b) Process the message to identify and extract the following message elements:

(1) Subject/Message ID
(2) Date-Time-Group (DTG)
(3) Message Action Addressee(s)
(4) Message Information Addressee(s)
(5) Office Symbol(s) (including multiple office symbols in a single address)
(6) Message Originator

(c) Extract U.S. classification and caveat markings including, but not limited to: TOP SECRET, SECRET, CONFIDENTIAL, UNCLAS EFTO FOUO, UNCLAS EFTO, and UNCLAS.

(d) Identify message handling markings for control functions used to enforce user "need-to-know" access within the various classification and caveat levels including, but not limited to: LIMDIS, NOFORN (or NFD), RESDAT, FORMERLY RESDAT (in all formats: FRD, etc.), EYES ONLY (and who for), PERSONAL FOR (and who the message is personal for), SPECAT, SPECAT EXCLUSIVE, and specified, special compartmental caveats as may be modified by the user.

3.2.2 MESSAGE OUTBOUND PROCESSING

Outbound processing consist of message generation, data validation, routing, release coordination, and handing off data to an external process/COE module for further processing. A flow chart depicting the association of these processes can be seen in Figure 3-2.

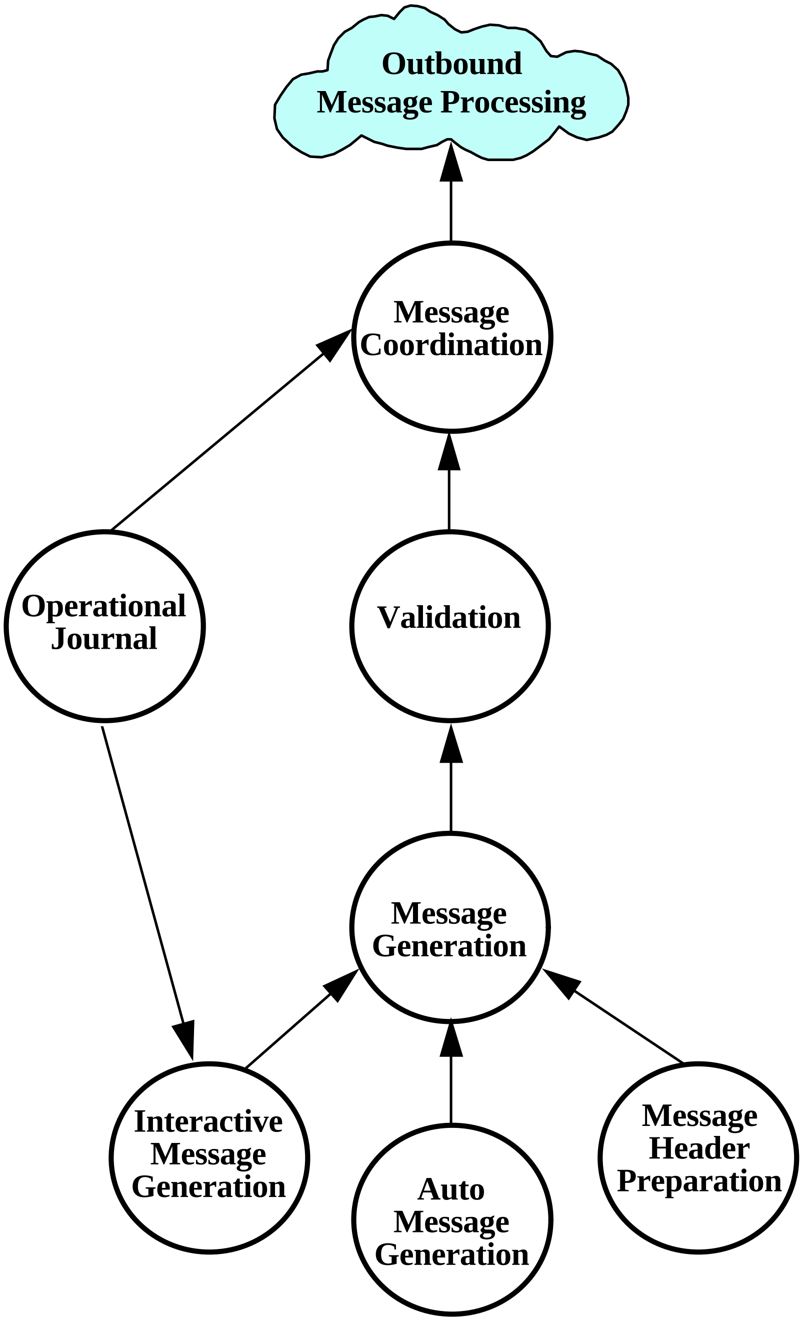

Figure 3-2 Message Outbound Processing

3.2.2.1 MESSAGE GENERATION

The message processing module must support generation of messages, regardless of the origin of the standard, as long as the message definition provided is in conformance with the MTF definition rules. Message generation is broken into two separate and distinct areas, automatic and interactive, with areas sharing common functionality. Regardless of the method used for message generation, the message processing module shall provide a means to submit messages, to the communications system, by precedence where the highest precedence is processed first. Rank order of precedence is:

(a) Emergency Command Precedence (ECP)
(b) Flash
(c) Immediate
(d) Priority
(e) Routine

3.2.2.1.1 MESSAGE AUTO GENERATION

Automatic message generation may be activated by multiple means, some internal to the message processing module while others are external. An examples of an internal process that may activate automatic message generation would be satisfaction of a SRI where the user has directed that satisfaction of a SRI criteria requires auto generation and output of a message to an external source. An example of an external activator, for automatic message generation, could be satisfaction of a Data Base (DB) SRI where the user has directed the system to create a message based upon specified activity occurring in the DB. In either case, the automatic message generation module requires that certain data be provided, such as format, addressee(s), message type, and adequate data to populate mandatory fields. The message processor shall be able to automatically generate messages for release based on:

3.2.2.1.1.1 Data routed to auto generation as the result of satisfaction of a SRI

3.2.2.1.1.2 Data routed to auto generation as the result of a query, either at the direction of a process or a user

3.2.2.1.1.3 Data routed to auto generation as the result of user request based on selection of an icon, symbology, and/or feature object from a map display

3.2.2.1.2 INTERACTIVE GENERATION

Interactive message generation relies upon input from a user to complete the message build. The process will be supported by some automated action(s) for fill of common fields with default information but will rely upon the user to supply data fill to areas which contain user/system “information”.   The message processor shall:

3.2.2.1.2.1 Provide an annotated form (template) for interactively constructing each of the message sets specified in MIL STD 6040 and supported service unique standards, as specified in paragraph 3.2.

3.2.2.1.2.2 Provide on-line and interactive help (context sensitive) in preparation of a message as provided by the electronic data representation of the associated message standard, when it exists as part of the message definition contained in the USMTF CDBS or appropriate format.

3.2.2.1.2.3 Support interactive message generation where input of any message field is supported in any combination of the following:

(a) User via keyboard entry and/or edit
(b) User via cut and paste
(c) Query results
(d) Validated data entry tables

3.2.2.1.2.4 Provide default values for message fields or sets which are user-definable or selectable.

3.2.2.1.2.5 Provide a capability for forwarding a message for coordination.

3.2.2.1.2.6 Provide the user the capability to edit and reroute or submit a message for release.
3.2.2.1.2.6.1 Allow user to switch from full templete to filled in templete for editing.
3.2.2.1.2.7 Provide the user with message addressing parameters to allow message routing to a specific address

3.2.2.1.2.8 Provide the user with message addressing parameters to allow message routing to multiple addresses

3.2.2.1.2.9 Support interactive message generation where auto filled sets/fields are modifiable by the user.

3.2.2.1.2.10 Support interactive message generation where input is through cut and paste operations and copied and pasted from previous set/field entries in same message.

3.2.2.1.2.11 Support interactive message generation where input is through cut and paste operations and copied and pasted from previous set/field entries in another message.

3.2.2.1.2.12 Support interactive message generation where input is through cut and paste operations from text in another window

3.2.2.1.2.13 Support interactive message generation allowing the user to retrieve a message from a storage area and edit of that message.
**3.2.2.1.2.14 Support selection from a window messages for operational use into another window
**3.2.2.1.2.15 Provide wordprocessing functionality for free text fiels

3.2.2.1.3 FORMAT SELECTION

Messages contain communication system specific header data elements used for selection of a protocol necessary for communications between systems. The communications module relies on the message processing module to pass parameters, such as originator, addressee, DTG, classification, etc. to it in order to determine what message format was used in creation of the message and what protocol is required. The JANAP-128 and ACP-126 (modified) message formats are two of the most commonly used header definitions.

The message processor shall be capable of selectively supporting message generation using the formats defined in paragraph 3.3.2.7 and/or supplying a generic header capable of supplying header data elements required by the communications module, thereby making the header data entry area generic.

**3.2.2.1.3.1 Provide validation of header.

**3.2.2.1.3.2 Provide auto-fill of message header where possible using default values.
**3.2.2.1.3.3 Outbound message shall be saved to a standard diskette

3.2.2.2 MESSAGE COORDINATION AND RELEASE

There is more than one way to create a message. A message may get generated based upon a SRI satisfaction where the information which caused satisfaction of the SRI criteria is forwarded for automatic message generation and to the communications module for release or the combined effort of multiple user where the finished product should be reviewed and concurred to by all prior to release. Another scenario might be where only one individual has been granted the authority to release messages. To facilitate coordination and release requirements the message processing module shall:

3.2.2.2.1 Support serial coordination of a message.

3.2.2.2.2 Support parallel coordination of a message.

3.2.2.2.3 Allow the user to specify distribution of an internally coordinated message, either adhoc or via a user-defined distribution lists.

3.2.2.2.4 Provide the capability to create a list of message coordination and release personnel.

3.2.2.2.5 Provide the capability to maintain a list of message coordination and release personnel.

3.2.2.2.6 Provide the capability to delete a list of message coordination and release personnel.

3.2.2.2.7 Provide the capability to notify the members on a list of message coordination and release personnel when they have a message awaiting review.

3.2.2.2.8 Allow users to annotate a message in coordination with their comments and route those comments back to the drafter of the message.

3.2.2.2.9 Provide the capability for members on the message coordination and release list to edit a message under review.

3.2.2.2.10 Distribute an internally generated message according to a routing list specified by the user.

3.2.2.2.11 Notify users when they have a message awaiting coordination.

3.2.2.2.12 Allow users to view the status of a message during the coordination cycle.

3.2.2.2.13 Allow users to receive notification when a suspense has been missed.

3.2.2.2.14 Authenticate releaser or release processes against a list of authorized message releasers.

3.2.2.2.15 Allow the releaser to reject the message back to the drafter.

3.2.3 MESSAGE PROCESSING SUPPORT SERVICES

The message processing module is made up of many stand alone and callable smaller modular. These modules can be, and are, used for both inbound and outbound processing. This architecture supports the concept of “sizing” by allowing the using system to select only modules which are necessary to perform functions desired by their user. The following paragraphs, along with Figure 3-3, describe support services.

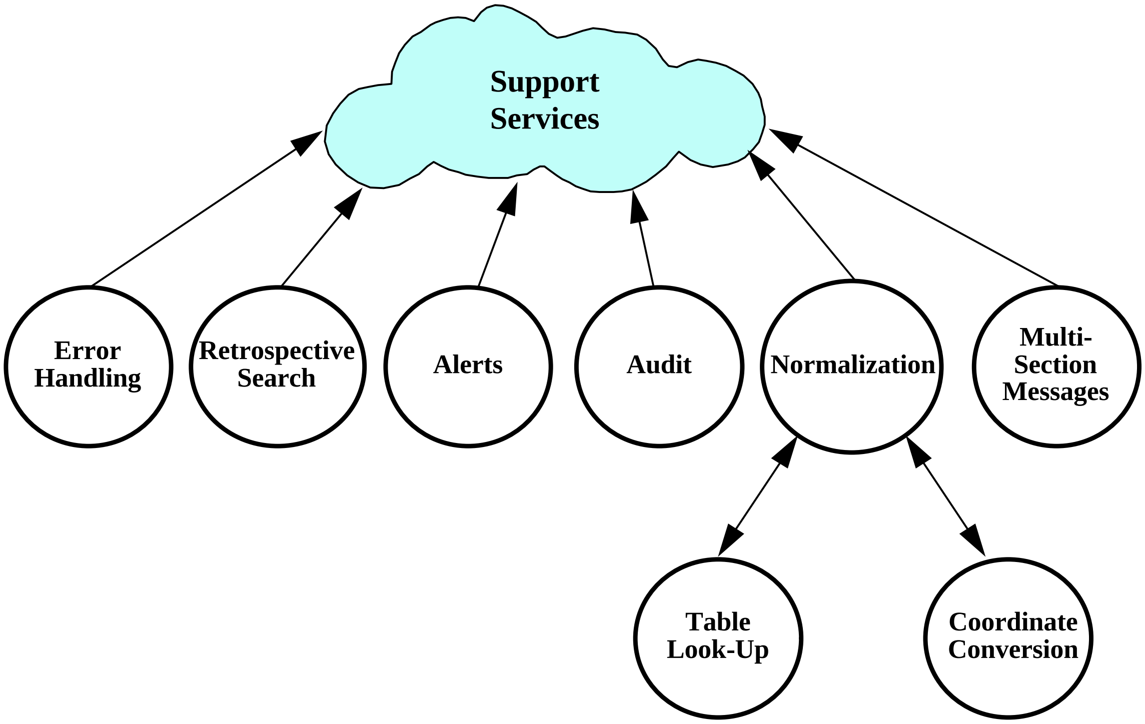

Figure 3-3 Message Processing Support Services

3.2.3.1 SYSTEM CONFIGURATION

System configuration, for the message processing module, concerns selection a routing table to be used for this session or message definition file (the system may have multiple message standard baselines loaded on the system but only one in use at any given time). At system start up the message processor will default to the configuration present when last terminated. The system administrator may change configuration items active at any time without adversely impacting other system resources and have the change implemented upon the next system call to the configuration item changed.

3.2.3.1.1 START UP

3.2.3.1.1.1 At start up, the message processor shall default to the configuration present at last termination

3.2.3.1.1.2 The system administrator shall be allowed to change configuration upon validation of the administrator’s access rights/privileges

3.2.3.1.2 TERMINATION

The message processor shall save its current configuration at termination in order to be restarted.

3.2.3.2 ERROR HANDLING

The message processor must be capable of accepting error conditions from internal and external sources. While error conditions will occur internally, failure of a message to pass validation checking, the message processor must also be capable of handling external errors, such as rejection of data passed by it to another module. One such source could be the communications module if the communications module failed to recognize a Plain Language Address (PLA) on a message passed to it. To enable error handling the message processor shall:

3.2.3.2.1 Provide for presentation of the error to a user for action

3.2.3.2.2 Provide the capability for an authorized operator to review (and override) the rejection of messages due to validation and/or verification errors.

3.2.3.2.3 Provide the capability to suspend the processing of messages received with validation and verification errors pending user review and correction.

3.2.3.2.4 Provide the capability for parser to resume processing of a message with errors after user review and correction.

3.2.3.2.5 Accept an error condition from the communications module when
(a) the message contains an error(s)
(b) the message transmission is abnormally terminated

3.2.3.3 AUDIT

The message processing module shall support audit requirements as specified in paragraph 3.8 and the Security Administration SRS.

3.2.3.4 RETROSPECTIVE SEARCH

Users of the message processing module must have the capability to retrieve data from the operational journal. Retrieval is required to support message generation, such as changing a minimum amount of information in an existing message and transmit it as a new one, or the user needs to do some research prior to generating a message. In addition, the message processor shall:

3.2.3.4.1 Provide users the capability to search the message journal retrospectively for the messages of interest based on a defined search criteria against the complete message and any annotations. The search criteria shall be definable using a SQL based language.

3.2.3.4.2 Provide a process the capability to search the message journal retrospectively for the message(s) of interest based on a defined search criteria against the complete message and any annotations. The search criteria shall be definable using a SQL based language.

3.2.3.4.3 Provide the user the ability to enter search criteria (SQL statement) interactively via a user interface.

3.2.3.4.4 Provide the user the ability to enter delivery information interactively via a user interface.

3.2.3.4.5 Have no set limit on the maximum number of searches that can be run against a particular message.

3.2.3.4.6 Have no set limit on the maximum number of users or processes to which results of a search can be sent.

3.2.3.4.7 Support modification of an old message therefore creating a new message by

(a) Retrieve and display a whole message

(b) Editing of the displayed message, to include format lines

(c) Replacement of the existing addressee(s)

(d) Allow the revised message to be submitted as a new message

3.2.3.5 NORMALIZATION

Normalization is the process by which data is transformed from one representation to a second form. In the case of normalization for an incoming message, data normalization is required to change the data representation of data in the message to one usable by the host system. Using this definition one can see a requirement to normalize data during message generation also. To support data normalization, the message processor shall:

3.2.3.5.1 Provide the capability to normalize the data in a received message to a form desired for use within the using system.

3.2.3.5.2 Normalize data from an inbound message to a form required by the application system in real or near-real time via a normalization algorithm and/or alias tables.

3.2.3.5.3 support coordinate conversion as provided by a coordinate conversion algorithm.

3.2.3.6 BOM TO COM CONVERSION

The message processing module is required to process Bit-Oriented-Messages (BOM) in addition to Character Oriented Messages (COM). To support this requirement the message processor shall:

3.2.3.6.1 Provide the ability to translate a BOM message to COM for continued processing, or

3.2.3.6.2 Provide the ability to parse a BOM message, or

3.2.3.6.3 Provide a path by which a BOM message may be passed directly to system resources for additional processing

3.2.3.7 Message Data Tables

Message data tables shall be constructed automatically from the MTF CDBS and or data bases which conform the CDBS scheme. The message processor does not care what message standards are supported in the message data table as long as those proposed conform to the MTF definition rules. Given that all information required by the CDBS like data base is available, the message definition tables will contain all the information required to parse, generate, validate, and provide on-line help for messages processed by the module. The message processor shall:

3.2.3.7.1 Allow the dynamic modification of message processor tables, by an authorized user, to:

(a) create new alias table entries
(b) modify data field contents (extend valid entry/code list)
(c) dynamic definition of database element definitions (e.g.; data normalization)

3.2.3.7.2 Provide the capability to update 100 percent of the supported messages from the CDBS through generation, and replacement, of new message definition data tables.

3.2.3.8 MESSAGE VALIDATION

Validation occurs for both inbound and outbound message processing, the difference being that on inbound processing validation is optional and on outbound processing it is mandatory (but may be over-ridden through manual intervention). Validation on inbound messages relies upon the user to determine the amount of validation, if any, to occur. If the user elects to validate the messages and errors are detected further processing shall either be inhibited until the error is resolved or errors marked such that additional processing of that message shall not extract and submit as new field already used to create duplicate records for output. For inbound validation the message processor shall:

3.2.3.8.1 Provide the capability to validate operational or exercise markings against the operational state.

3.2.3.8.2 Perform validation for those message formats outlined in 3.3.2.7

3.2.3.8.3 Validate for correct format, content and conditionally in accordance with approved format tables and data entry code lists for message format types outlined in 3.3.2.7

3.2.3.8.4 Route messages with detected validation errors to a user for interactive correction.

3.2.3.8.5 Provide the user with aides to correct message validation errors the system identifies.

3.2.3.8.6 Validate that all message mandatory sets/fields exist

3.2.3.8.7 Validate that mandatory set/field contain allowable values

3.2.3.8.8 Validate use of optional and conditional/fields sets

3.2.3.8.9 Validate that optional and conditional set/field contain allowable values

3.2.3.8.10 Determine that the set contains all mandatory fields.

3.2.3.8.11 Determine that the set contains no more than the specified maximum number of fields

3.2.3.8.12 Determine that the set sequence ordering conforms to the standard specification.

3.2.3.8.13 Determine that the segment ordering conforms to the standard specification.

3.2.3.8.14 Determine whether all special instructions specified in the standard are followed in the message.

3.2.3.8.16 Provide for a manual over-ride of identified validation errors for outbound messages

3.2.3.8.17 PERFORM AUTOMATIC VALIDATION CHECK AFTER EDITING MESSAGE

3.2.3.9.18 ABILITY TO VALIDATE MESSAGE TO CURRENT APPROPRIATE VALIDATION TABLE

3.2.3.9 MULTI SECTIONED MESSAGES

The message processor must be capable of supporting the communications module requirement that packets released to it for transmission conform to size limitations. Conformance is normally enforced through message segmentation. To support message segmentation the message processor shall provide the capability to section a message into segments, having a maximum length of 40,000 bytes per segment. Additionally, the message processor shall:

3.2.3.9.1 Create a single message from all sections of a multi-part message received in an operator setable time period.

3.2.3.9.2 Process an incomplete sectioned message at user discretion/direction.

3.2.3.9.3 Ensure that the message is displayed in section order regardless of whether sections are received out-of-sequence or contains missing sections.

3.2.3.9.4 Place an indicator in the reconstituted message where portions of the message/message text are missing.

3.2.3.10 MESSAGE ANNOTATION

Message coordination, parsing, and retrospective search are possible processes requiring the user to attach comments to a message stored in the operational message journal. For sure, reviewers of a message will want to comment on the message preparation/content and/or attach a sign-off on messages prior to release. To support message annotation, the message processor shall:

3.2.3.10.1 Provide a mechanism whereby memorandums may be attached to a base record

3.2.3.10.2 Insure that attached comments are not released along with the base record

3.2.3.10.3 Place no limitation on the number of memorandums which may be attached to any one base record

3.2.3.11 MESSAGE RETRANSMISSION

The message processing module must appropriately mark messages retrieved during retroactive search and reintroduced into a network/net for transmission to another destination. An example of this could be generation and release of a unit status message where there is minor changes from day to day. The user requires the ability to retrieve a previous message, modify it, and reintroduce the message into the system as a new message. Also, a higher unit may want to retransmit a message to a lower echelon, for information purpose, and only add that user as an addressee, virtually unchanging the original message. The message processing module shall:

3.2.3.11.1 Release messages with changes in the "TO" and/or "INFO" lines and no changes to text in a re-addressal format.

3.2.3.11.2 Release messages with no changes to text as corrected copy (i.e. ZDK as ZZS is utilized when an error is made by the serving communications center).

3.2.3.11.3 Mark messages which have been retrieved from a storage area that have been acknowledged as delivered but are reintroduced with no changes as exact duplicates (ZFG).

3.2.3.11.4 DELETED

3.2.3.12 OPERATIONAL JOURNAL

The message processor shall provide an area where both inbound and outbound messages and temporary storage of messages under construction or in coordination are filed. Records placed in the operational journal will be accessible by authorized users for retrieval, annotation, and/or additional processing. The operational journal shall:

3.2.3.12.1 Provide the capability to generate/display a directory of journal records

3.2.3.12.2 Provide the capability to log the following information from received and/or transmitted messages:

(a) Date and time of message origination - Date Time Group (DTG)
(b) Date and time the message was received
(c) Subject/Message ID
(d) Message originator
(e) Message destination
(f) Security classification (including codewords/nicknames and handling caveats)
(g) Message identification and number.

3.2.3.12.3 Provide the capability to selectively log the following information from received and transmitted messages at the request of an application program:

(a) Message sender
(b) Releaser
(c) Message type
(d) Message transmission status

3.2.3.12.4 Provide the capability to selectively retrieve the logged message information.

3.2.3.12.5 Provide the capability to maintain the status of all messages under coordination and release review.

3.2.3.12.6 Provide the capability for members on the message coordination and release list to access the status of all messages in coordination.

3.2.3.12.7 Provide the capability to store the contents of the receive queue for subsequent retrieval in the event of a W/S re-initialization.

3.2.3.12.8 Provide the capability to store selected messages on-line for quick access. Storage parameters shall include the message profile (i.e., type, originator, precedence) and specified time period (i.e., for 1, 2, 6, 12, 24, and 48 hours).

3.2.3.12.9 Provide the capability to search the message storage for messages of interest based upon a user-defined set of search criteria.

3.2.3.12.10 Provide the capability to define message storage space capacity and threshold depletion limits.

3.2.3.12.11 Provide the capability to monitor message storage space for depletion.

3.2.3.12.12 Provide the capability to route messages to an alternate, selectable, device when the message storage area limits are reached.

3.2.3.12.13 Provide the capability to disable the acceptance of incoming non-ECP and non-Flash messages when primary storage devices are full

3.2.3.12.14 Provide the capability to delete the oldest messages having the same or lower precedence as the incoming message in order to make room when the message storage area is full and acceptance of incoming messages is not disabled.

3.2.3.12.15 Provide the capability to enable the acceptance (for storage) of incoming messages according to message precedence.

3.2.3.12.16 Provide the capability for an application program to delete selected messages from the message storage area.

3.2.3.13 PERFORMANCE REQUIREMENTS

3.2.3.13.1 For UNIX based systems, the message processing module shall meet the performance goals specified below:

(a) The message processing module should meet user responsiveness times as specified in Table 3-1.

Function	Criteria
From the time that the user orders its formulation, display or make available for display by scrolling or paging the first page of a summary display.	50% in 1.5 sec 95% in 2.0 sec 100% in 2.5 sec
In response to a command for a general database search, assemble, order, and format a summary display from the message database and transfer and display the first page.	50% in 3.0 sec 95% in 4.0 sec 100% in 5.0 sec
Reorder a summary display, such as rearranging the sequence, adding or deleting message summary fields, or producing a display containing a subset of the original display.	50% in 1.0 sec 95% in 2.0 sec 100% in 2.5 sec
Retrieve and display the first page of a message stored in the system database in response to user interaction with the associated message summary display.	50% in 1.5 sec 95% in 2.0 sec 100% in 2.5 sec
Display the first page of a message stored off-line after acknowledgment that off-line media must be transferred to an on-line device.	10 min

Make available for display, by scrolling or paging, the next or preceding page of an object, after the current page has been viewed for one second.	1.0 sec
Scroll rate.	20 lines/sec
Error feedback following completion of an input.	3.0 sec
Response to simple command (e.g., delete message).	1.0 sec
Response to complex command (e.g., advise a user that a requested message has been archived).	4.0 sec

Table 3-1 User Responsiveness Performance Criteria

(b) The message processing module should meet message in-processing times as specified in Table 3-2.

Precedence	Time (Sec)
95% of ECP messages	10
100% of ECP messages	30
95% of Flash messages	15
100% of Flash messages	60
95% of Immediate messages	30
100% of Immediate messages	120
95% of Priority messages	60
100% of Priority messages	240
95% of Routine messages	120
100% of Routine messages	500

Table 3-2 In-Processing Performance Requirements

(c) The message processing module should meet message out-processing times as specified in Table 3-3.

Precedence	Time (Sec)
95% of CRITIC messages	10
100% of CRITIC messages	30
95% of ECP messages	10
100% of ECP messages	30
95% of Flash messages	20
100% of Flash messages	60

95% of Immediate messages	60
100% of Immediate messages	240
95% of Priority messages	120
100% of Priority messages	500
95% of Routine messages	360
100% of Routine messages	1000

Table 3-3 Out-Processing Performance Requirements

3.2.3.13.2 For Windows/DOS based systems, the message processing module shall meet the performance goals as stated in the WINDOWS based message generation requirements document.

3.3 MESSAGE PROCESSING EXTERNAL INTERFACE REQUIREMENTS

3.3.1 MESSAGE PROCESSING INTERFACES

The message processing module interfaces with the COE-supplied Communications module to receive messages for further processing and hands off new messages to the communications module for transmission. Note that the CMP is currently coordinating and planning for interface with the Defense Messaging System (DMS) as the future communications front end. The message processing module also interfaces with processes, which include COE supplied system services, DBMS, user interaction, alerts, etc., for passing processed message information to the system for action. The message processor receives data from system processes for use in message generation. Internally, the message processor interfaces to low level modules to accomplish the requirements listed above.

3.3.2 MESSAGE PROCESSING INTERFACE IDENTIFICATION

3.3.2.1 The message processing module shall interface with the communications area for receipt of and hand off of messages for transmission to external systems

3.3.2.2 The message processing module shall interface with the security administration software for receipt of access control and/or user privilege information

3.3.2.3 The message processing module shall interface with the audit software for storage and manipulation of audit information

3.3.2.4 The message processing module shall alternatively provide an interface to a text processing subsystem to generate freetext messages.

3.3.2.5 The message processing module shall alternatively provide an interface to an Office Automation e-mail subsystem for message distribution and introduction of messages e-mail messages for release.

3.3.2.6 The message processing module shall interface with COE system and application support modules to receive and transmit messages. Specific modules include:

(a) Queuing mechanisms
(b) Distributed Computing Environment (DCE) modules
(c) Alert services
(d) Security services
(e) Database administration

(f) MCG&I services
(g) Office automation software
(h) Communications support software

3.3.2.7 As appropriate, the message processing module shall receive, generate, validate, and distribute the following communications message formats/standards:

(a) ACP-126
(b) ACP-126 (modified)
(c) ACP-127
(d) JANAP 128
(e) MTS
(f) DOI-103
(g) ACP-123
(h) DD173
(i) ACP-127 (modified)
(j) IEWCOMCAT
(k) ULP

3.3.2.8 The message processing module shall provide an interface to an on-line message storage device

3.3.2.9 The message processing module shall interface with functional applications in order to deliver message data.

3.4 MESSAGE PROCESSING INTERNAL INTERFACE REQUIREMENTS (TBS)

3.5 MESSAGE PROCESSING INTERNAL DATA REQUIREMENTS

All message processing components shall be automatically derived from an electronic data representation of the associated message standard, when it exists.

3.6 Adaptation Requirements

This paragraph has been removed through tailoring of the Data Item Description.

3.7 Safety Requirements

This paragraph has been removed through tailoring of the Data Item Description.

3.8 Security and Privacy Requirements

Security policy enforcement is the responsibility of the Trusted Computing Base (TCB). The COE design assumes that COE Layers 1 and 2 will contain COTS /GOTS products adhering to either the C2 or B1 levels of operational requirements, as defined in the Trusted Computer System Evaluation Criteria. Satisfaction of security requirements needed to adhere to the COE Security Policy are allocated to trusted components. Processes that can be implemented without exemption from security controls will be labeled as untrusted. Untrusted code is not responsible for enforcing security, but must follow the policy enforced by the TCB. The resulting requirements on untrusted code are derived from the COE security policy.

The objective COE will be integrated into systems intended to be evaluated at the B1 or higher evaluation class. Guidelines for developing trusted and untrusted software should be followed to ease the eventual migration to the multilevel secure system required by DoD. Development guidelines for untrusted and trusted software, respectively, are documented in DoD 5200.28-STD series of documents.

3.8.1 TRUSTED SOFTWARE REQUIREMENTS

Exactly which COE components must be trusted can only be determined based on the COE security architecture. It is the responsibility of the system and application developer to determine how trusted and untrusted components are integrated. The requirements below address functionality that must be trusted in order to meet COE security processing requirements.

3.8.1.1 If the message processor is responsible for appending information labels based upon “actual classification labeling” vice system higher water mark, then it shall demonstrate compliance with the B1 evaluation class in a manner that provides data integrity and security protection as defined in DoD 5200.28-STD.

3.8.1.2 If the Message-Based SRI module evaluates and routes based on classification levels, and is responsible for trusted output then it shall demonstrate compliance with the B1 evaluation class in a manner that provides data integrity and security protection as defined in DoD 5200.28-STD.

3.8.1.3 In a system using target COE software/modules and providing C2 security, the Message Set Classification public operation shall provide advisory security labels in support of manual downgrade of messages.

3.8.1.4 In a system using target COE software/modules and providing C2 security, the Message Get Classification public operation shall provide advisory security labels in support of manual downgrade of messages.

3.8.1.5 In a system using objective COE software/modules and providing B1 security, the Message Set Classification process shall be implemented in a trusted process and shall be valid only when invoked by a trusted subject.

3.8.2 UNTRUSTED SOFTWARE REQUIREMENTS

Untrusted software is impacted by security enforcement imposed by the TCB. The first element of security enforcement is Mandatory Access Control (MAC) on information flow between components: Multilevel security is transparent to untrusted software in that untrusted code has no knowledge of security labels maintained by the TCB. However, MAC in multilevel secure systems restricts data flow between system components. This will impact the way in which various trusted and untrusted components are integrated into the system's architecture. For example, all users of an untrusted application may be required to operate at an application-high security level. The second element of security enforcement is the restriction of privileges for individual components. Untrusted software uses only the standard operating system and supports software services that require no special privileges.

In addition to the security enforcement imposed by the TCB, a secure system provides a small selection of security features that are visible and available to untrusted software. Where appropriate, COE provides interfaces to these features.

The following are general processing requirements for untrusted Message Processing Area:

3.8.2.1 Any distinct untrusted processes in the Message Processing Area (e.g., functions not linked into application code) that communicate with one another shall run at the same security level.

3.8.2.2 Untrusted software shall use only the standard operating system and support software services that require no special privileges.

3.8.2.3 If the underlying COTS software provides security features that are visible to untrusted applications, then untrusted COE components shall make available an interface to those features.

3.8.3 ACCOUNTABILITY

All transactions that occur within the message processor module, and those that occur between the message processor module and external modules will be accounted for. The method for providing this accountability is by use of an audit trail. To support this audit trail, the message processor module shall:

3.8.3.1 Output an audit record for each occurrence of a user definable transaction (user here refers to an authorized administrator with access to system configuration files and possible the audit trail).

3.8.3.2 Record the following with every audit record:

(a) Date and time of event
(b) Event
(c) Security markings
(d) Success or failure
(e) User ID
(f) Duty position/role (if applicable)

3.8.3.3 Record the following with every message-related audit record:

(a) Date and time of message origination (DTG)
(b) Subject/Message ID
(c) Message Originator
(d) Message Destination
(e) Security Classification (including codewords/nicknames and handling caveats)
(f) Message identification and number

3.8.3.4 Provide the capability to audit the following types of events:

(a) Beginning and ending of a message database session.
(b) Access to messages in the message processing module database.

3.8.3.5 Provide a protection mechanism for audit data such that read and modify access is limited to users with validated access rights/privileges.

3.8.4 ACCESS

Access rights/privileges shall be controlled and supplied by the Security Administration software

3.8.5 LABELS

Information labels are required to be attached to every object within a system, if that system is required to maintain a relationship between information within the system and the actual classification of the data (see paragraph 3.2.1.3 for additional information). If a system is to be evaluated and accredited to operate at the B1, or higher, level of certification the message processor shall:

3.8.5.1 Attach an information label(s) to each object created.

3.8.5.2 Create information labels IAW DIAM 65-19.

3.8.6 MARKING

Security marking shall be applied to all data when exported to a hardcopy device IAW DIAM 65-19.

3.8.7 DATA CONTINUITY

Data continuity shall be maintained through use of persistent queuing or method equally reliable in assuring that data is not inadvertently lost.

3.8.8 DATA INTEGRITY

Data integrity shall be retained through protection of data such that the data is not exposed to accidental or malicious alterations or destruction

3.8.9 OBJECT REUSE

Object reuse shall be in conformance with DoD 5200.28.

3.8.10 CONTINGENCY PLANNING

This paragraph has been removed through tailoring of the Data Item Description.

3.8.11 SYSTEM ARCHITECTURE

The message processor software module shall conform to the COE architectural design philosophy and constraints.

3.8.12 SYSTEM INTEGRITY

This paragraph has been removed through tailoring of the Data Item Description.

3.9 MESSAGE PROCESSING ENVIRONMENT REQUIREMENTS

3.9.1 SOFTWARE REQUIREMENTS

The Message Processor is intended for use across multiple hardware platforms and operating systems in support of the DoD implementation of the Defense Information Infrastructure (DII) Common Operating Environment (COE), a cost reduction strategy affecting development and maintenance of software and Interoperability.

Minimum software requirements for successful hosting on a UNIX system are:

UNIX Operating System (various implementations)
X11R5

Motif Windows manager (MIT version)
Communications front end for receipt and release of record traffic

Minimum requirements for hosting on a Windows environment are contained in the CMP message generation requirements document for Windows which is TBS.

3.10 MESSAGE PROCESSING RESOURCE REQUIREMENTS

3.10.1 HARDWARE REQUIREMENTS

COE software capabilities will be developed for the following platforms (A platform is a selected pairing of Computing Hardware and an Operating System):

(a) Army CHS product list
(b) Navy TAC IV product list
(c) Personal/stand alone computer with a Windows operating system

3.10.2 HARDWARE RESOURCE UTILIZATION REQUIREMENTS

Minimum hardware requirements for successful hosting of the current implementation on a UNIX platform are:

32 MB Random Access Memory (RAM)
30 MB hard disk space available

Minimum hardware requirements for successful hosting of the Windows message generation segment will be contained in the CMP message generation requirements document for Windows which is TBS.

3.11 SOFTWARE QUALITY FACTORS

This paragraph has been removed through tailoring of the Data Item Description.

3.12 DESIGN AND IMPLEMENTATION CONSTRAINTS

3.12.1 The message processing module shall provide upwardly compatible interfaces between COE versions of the Message Processing Area and application programs.

3.12.2 The message processing module shall provide upwardly compatible functional services of the Message Processing Area in COE versions to applications programs.

3.12.3 The message processing module shall be portable across Government Furnished Equipment/ Government Furnished Information (GFE/GFI) common hardware and software platforms.

3.12.4 The design and implementation of the message processing module shall conform to the COE architecture.

3.12.5 The message processing module shall incorporate an open systems architecture design, in accordance with that defined by the Defense Information Infrastructure (DII) Common Operating Environment (COE) to allow integration with other applications/systems.

3.12.6 The message processing module shall operate in a Distributed Computing Environment (DCE).

3.12.7 The message processing module shall allow for future growth/expansion and portability through early definition and stability of Application Programmer Interfaces (API).

3.12.8 The message processing module shall be designed to support operations of selected modules (those required for message generation) in a DOS/Windows environment.

3.13 COMPUTER COMMUNICATIONS REQUIREMENTS

3.13.1 The message processing module shall use the operating system and DCE services to route messages, journal messages, and parsed data to the appropriate directory or application.

3.13.2 The message processing module shall use the COE Communications Services Area to route messages across a Wide Area Networks (e.g., Mobile Subscriber Equipment (MSE) and Combat Net Radio (CNR)).

3.14 PERSONNEL-RELATED REQUIREMENTS

3.14.1 All message preparation instructions and help text shall be provided to the drafter/user by the message processing module.

3.14.2 The message processing module shall provide on-line and interactive help (context sensitive).

3.14.3 Service unique help shall be provided if adequate information is supplied by the services in a format compatible with USMTF CDBS standard.

3.14.4 The human interface shall be developed IAW the DII COE Style Guide.

3.15 TRAINING-RELATED REQUIREMENTS

This paragraph has been removed through tailoring of the Data Item Description.

3.16 LOGISTICS-RELATED REQUIREMENTS

This paragraph has been removed through tailoring of the Data Item Description.

4. QUALIFICATION PROVISIONS

The message processor and its message definition data tables require joint testing and approval by the Joint Interoperability Test Center (JITC) for USMTF standard compliance.

Requirement	Paragraph ID	Qualification
MESSAGE STATES AND MODES	3.1	Demo & Test
MESSAGE INBOUND PROCESSING	3.2.1	
INTERNAL ROUTING	3.2.1.1	Demo & Test
MESSAGE PARSER	3.2.1.2	Demo & Test
INFORMATION LABELING	3.2.1.3	Demo & Test
SRI PROCESSING	3.2.1.4	Demo & Test
MESSAGE PROFILING	3.2.1.5	Demo & Test
MESSAGE OUTBOUND PROCESSING	3.2.2	
MESSAGE GENERATION	3.2.2.1	Demo & Test
AUTO GENERATION	3.2.2.1.1	Demo & Test
INTERACTIVE GENERATION	3.2.2.1.2	Demo & Test
FORMAT SELECTION	3.2.2.1.3	Demo & Test
MESSAGE COORDINATION AND RELEASE	3.2.2.2	Demo & Test
MESSAGE PROCESSING SUPPORT SERVICES	3.2.3	
SYSTEM CONFIGURATION	3.2.3.1	
START UP	3.2.3.1.1	Demo & Test
TERMINATION	3.2.3.1.2	Demo & Test
ERROR HANDLING	3.2.3.2	Demo & Test
AUDIT	3.2.3.3	Demo & Test
RETROSPECTIVE SEARCH	3.2.3.4	Demo & Test
NORMALIZATION	3.2.3.5	Demo & Test
BOM TO COM CONVERSION	3.2.3.6	Demo & Test
MESSAGE DATA TABLES	3.2.3.7	Demo
MESSAGE VALIDATION	3.2.3.8	Demo & Test
MULTI SECTIONED MESSAGES	3.2.3.9	Demo & Test
MESSAGE ANNOTATION	3.2.3.10	Demo & Test
MESSAGE RETRANSMISSION	3.2.3.11	Demo
OPERATIONAL JOURNAL	3.2.3.12	Demo & Test

Table 4-1 Qualification Methods

5. REQUIREMENTS TRACEABILITY

TRACEABILIY AVAILABLE UNDER SEPARATE SOURCE.

6.0 REQUIREMENTS TRACEABILITY

TRACEABILIY AVAILABLE UNDER SEPARATE SOURCE

7. NOTES

Standard Verbs

A set of unambiguous transitive verbs has been identified and defined. The verbs have been used in the development of this functional requirements specification

ABORT	Terminating an activity prematurely.
ACCEPT/REJECT	Receiving data that is judged to satisfy a requirement, and the reverse.
ACCESS	Reading or writing data structures from a mass storage device.
ACKNOWLEDGE	Reporting the receipt of a message and whether the message was with, or without errors to the originator of the message.
ACTIVATE/DEACTIVATE	Causing a device to begin running, and the reverse.
ADD/MODIFY/DELETE	Manipulating/changing data elements.
ADDRESS	Providing a unique identifier for the receiver of data.
ALLOCATE/DEALLOCATE	Designating storage resources for a specific purpose, and the reverse.
ASSIGN	Giving out a task; delegating responsibility for an activity to a subordinate.
CLASSIFY/DECLASSIFY	Associating *a* DoD security classification to an element, and the reverse.
COMMUNICATE	Sending/receiving messages between logical and/or physical entities.
CONFIGURE	Identifying, and arranging the elements in a group or network.
CONNECT/DISCONNECT	Linking elements across a communications circuit, and the reverse.

CONVERT Changing a data element from one form or state to another.

CREATE/DESTROY Causing a data element to exist, bringing it into being, building it, or producing it, and the reverse.

CUT/COPY/PASTE Removing a selected data from a screen display, duplicating a selected data from a screen display, and redisplaying a previously cut or copied data on a screen display.

DEFINE Describing the precise nature and qualities of entities (e.g., of a data element, data storage).

DELAY Suspend processing for some specified finite period of time.

DETERMINE Evaluating or appraising based upon specific criteria or knowledge base.

DISPLAY Exhibiting a data element or group of elements on a visual data W/S.

DISTRIBUTE Dispersing data elements to identified local activities or across a network.

DOWNLOAD/UPLOAD Transferring data from a superior to a subordinate, and the reverse.

EDIT Correcting, modifying, or adapting a data element in a controlled manner.

ENABLE/DISABLE Allowing a designated activity to be performed, and the reverse.

ENFORCE Compelling observance of specified standard of practice.

ENSURE Performing a decisive action to achieve a desired result.

ENTER Introducing a data element into the system from an outside source.

ERASE Replacing all information in a designated storage area with binary ones.

EXCHANGE Transmitting data and receiving data in return between logical or physical entities.

EXPAND/CONTRACT Increasing or decreasing size (e.g., computer resource, data structure).

EXPOSE/HIDE Making data elements on a visual data W/S visible, and the reverse.

FILL Entering data into pre-defined storage structures (e.g., forms).

FIND/SEARCH Locating a data element of a designated value or a set of values.

FORMAT 1. Transferring application-specific information (e.g., map regions military symbology, text, etc.) into a form understandable by the underlying graphics package

2. Initializing certain storage media.

FORWARD Sending received data on to a subsequent destination or address.

HANDLE	Accessing, controlling, or releasing a data element.
HIGHLIGHT/DIM	Making a data element prominent by altering its visual representation, and the reverse.
IDENTIFY	Ascertaining the identity and/or the nature of a data element.
IMPLEMENT	Proceeding according to a plan or design.
INITIALIZE	To load, and/or make ready to execute, and/or execute in order to establish a set of starting conditions.
INITIATE/TERMINATE	Causing a designated activity or process to begin, and the reverse.
INPUT/OUTPUT	Getting data from a device (not storage) or activity, and the reverse.
LABEL	Applying an annotation to the designated data element.
LIMIT	Restricting the value of a data element to pre-defined boundaries.
LOCK/UNLOCK	Restricting access to data elements or storage areas, and the reverse.
LOG	Recording/printing designated events and selected related information.
MAINTAIN	Preserving designated data elements through correction and updates.
MERGE	Combining sorted data retaining the original ordering scheme.
MONITOR	Systematically watching for the occurrence of designated events or data.
NOTIFY	Returning a message to a designated activity or person.
OPEN/CLOSE	Making the contents of a file visible and accessible, and the reverse.
OPERATE	Functioning effectively according to pre-defined rules.
PARSE	Breaking a compound data element down into components.
POLL	Interrogating a server to assess status, determine availability of data.
POSITION	Placing a data element in the desired location on a display.
PREVENT	Performing a decisive counteraction to stop something from happening.
PROCESS	Following a series of operations that bring about a result.
PROVIDE	Furnishing or giving access to a designated capability or service.
PURGE	The procedure to totally and unequivocally erase or overwrite all information stored in memory. Purging is one prerequisite to declassification of media. (Purging is performed on an entire media basis.)
QUALIFY	Meeting specified requirements.

QUEUE/DEQUEUE Adding an entry (data element) to a queue, or removing an entry from the queue.

READ/WRITE Getting data from a mass-storage device, and the reverse.

RECONFIGURE Changing or rearranging the elements in a group or network.

REFORMAT Changing the organization of a data element from one form to another.

REINITIALIZE Redefining the starting conditions of an activity and restarting it.

REPOSITION Moving a data element from one location to another on a display.

RETRIEVE Finding and bringing back, usually by copying the desired entity.

RETURN
1. Passing data elements to a requesting application program
2. Going back to a predefined location or configuration.

ROUTE Providing a message destination and/or transmission path.

SANITIZE Removing selected information for the purpose of changing the classification of a file or object from one classification to another

SECTION To divide or segment a message into fragments.

SELECT/DESELECT Choosing from a number of pre-defined alternatives, and the reverse.

SEND/RECEIVE Transmitting data over a communication link, and the reverse.

SET Changing the designated data element to the desired value or state.

STORE Transferring data to a specified storage media.

SUBMIT Entering a request.

SUSPEND/RESUME Interrupting an activity with the possibility of restart, and the reverse.

TRANSFER Conveying or shifting a data element or message from one location to another.

TRANSFORM Converting data from one representation to another.

UPDATE Changing the content of a data element to provide replacement information.

UTILIZE Employing the services or functionality of some other specified capability.

VALIDATE Determining whether a data element should receive official sanction.

VERIFY Determining whether a data element meets pre-defined criteria.

WAIT Suspend processing until one or more events occur.

7.1 INTEGRATION APPROACH AND ASSESSMENT OF RISK

7.2 PRIORITIZED LIST OF FUNCTIONAL CAPABILITIES

APPENDIX A WINDOWS NT REQUIEMENTS